

## Consultas SQL con condicionales

Para los ejemplos indicados a continuación, utilice la base de datos Northwind, generada en el recurso anterior.

En este documento usted encontrará:

Consultas SQL con condicionales

La sentencia CASE de SQL

COALESCE

IFF

Funciones SQL Null

¿Qué es un valor NULO?

IS NULL

El operador IS NOT NULL

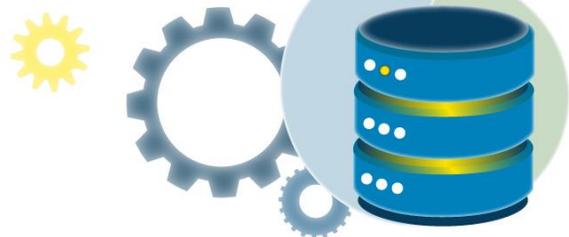
Función ISNULL ()

### La sentencia CASE de SQL

La sentencia **CASE** pasa por condiciones y devuelve un valor cuando se cumple la primera condición (como una declaración **if-then-else**). Entonces, una vez que una condición es verdadera, dejará de leer y devolverá el resultado. Si no se cumple ninguna condición, devuelve el valor de la cláusula ELSE. Si ninguna condición es verdadera, devuelve **NULL**.

#### Sintaxis CASE

```
CASE
  WHEN condition1 THEN result1
  WHEN condition2 THEN result2
  WHEN conditionN THEN resultN
  ELSE result
END;
```



## Ejemplos de CASE SQL

El siguiente SQL pasa por condiciones y devuelve un valor cuando se cumple la primera condición:

### Ejemplo

```
SELECT OrderID, Quantity,  
CASE  
    WHEN Quantity > 30 THEN 'The quantity is greater than  
30'  
    WHEN Quantity = 30 THEN 'The quantity is 30'  
    ELSE 'The quantity is under 30'  
END AS QuantityText  
FROM OrderDetails;
```

El siguiente SQL ordenará a los clientes por ciudad. Sin embargo, si la ciudad es NULL, ordene por país:

### Ejemplo

```
SELECT CustomerName, City, Country  
FROM Customers  
ORDER BY  
(CASE  
    WHEN City IS NULL THEN Country  
    ELSE City  
END);
```

## COALESCE

La función COALESCE () devuelve el primer valor, no nulo en una lista.

### Sintaxis

```
COALESCE(val1, val2, ..., val_n)
```

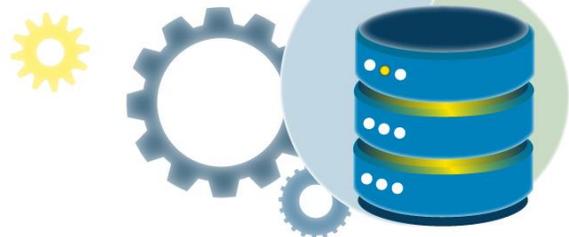
### Parámetros

Parámetro	Descripción
<i>val1, val2, val_n</i>	Requerido. Valores para testear

### Ejemplo

Devuelve el primer valor, no nulo de una lista:

```
SELECT COALESCE(NULL, NULL, NULL, 'W3Schools.com', NULL, 'Ex  
ample.com');
```



## IIF

La función IIF () devuelve un valor si una condición es VERDADERA, u otro valor si una condición es FALSA.

### Sintaxis

```
IIF(condition, value_if_true, value_if_false)
```

### Parámetros

Parámetro	Descripción
<b>condition</b>	Requerido. Valores para testear.
<b>value_if_true</b>	Opcional. Valor para retornar, si la condición es VERDADERA.
<b>value_if_false</b>	Opcional. Valor para retornar, si la condición es FALSA,

### Ejemplo

Devuelve 5 si la condición es VERDADERA, o 10 si la condición es FALSA:

```
SELECT IIF(500<1000, 5, 10);
```

Pruebe si dos cadenas son iguales y devuelva "SÍ" si lo son, o "NO" si no:

```
SELECT IIF(STRCMP("hello", "bye") = 0, 'YES', 'NO');
```

Devuelve "MÁS" si la condición es VERDADERA, o "MENOS" si la condición es FALSA:

```
SELECT OrderID, Quantity, IIF(Quantity>10, 'MORE', 'LESS')  
FROM OrderDetails;
```

## Funciones SQL Null

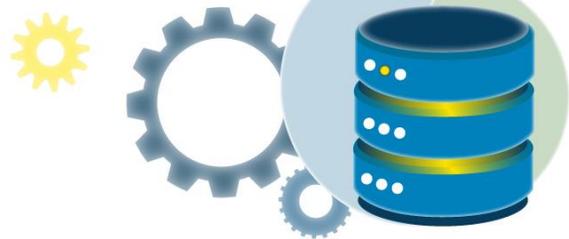
### ¿Qué es un valor NULO?

Un campo con un valor NULL, es un campo sin valor. Si un en una tabla es opcional, es posible insertar un nuevo registro o actualizar un registro, sin agregar un valor a este campo. Luego, el campo se guardará con un valor NULO.

**Nota:** Un valor NULO es diferente de un valor cero o un campo que contiene espacios. Un campo con un valor NULL es uno que se ha dejado en blanco durante la creación del registro.

### ¿Cómo probar los valores NULL?

No es posible probar valores NULL con operadores de comparación, como =, < >. Tendremos que usar los operadores **IS NULL** y **IS NOT NULL** en su lugar.



#### Sintaxis IS NULL

```
SELECT column_names  
FROM table_name  
WHERE column_name IS NULL;
```

#### Sintaxis IS NOT NULL

```
SELECT column_names  
FROM table_name  
WHERE column_name IS NOT NULL;
```

#### IS NULL

El operador **IS NULL** se utiliza para probar valores vacíos (valores NULL).

El siguiente SQL enumera todos los clientes con un valor NULL en el campo "Dirección":

#### Ejemplo

```
SELECT CustomerName, ContactName, Address  
FROM Customers  
WHERE Address IS NULL;
```

**Consejo:** use siempre IS NULL para buscar valores NULL.

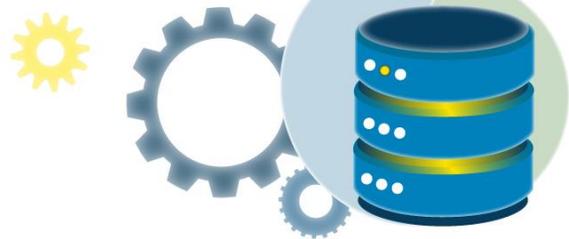
#### El operador IS NOT NULL

El operador **IS NOT NULL** se emplea para probar valores no vacíos (valores NO NULOS).

#### Ejemplo

El siguiente SQL enumera todos los clientes con un valor en el campo "Dirección":

```
SELECT CustomerName, ContactName, Address  
FROM Customers  
WHERE Address IS NOT NULL;
```



## Función ISNULL ()

Mire la siguiente tabla "Productos":

P_Id	ProductName	UnitPrice	UnitsInStock	UnitsOnOrder
1	Jarlsberg	10.45	16	15
2	Mascarpone	32.56	23	
3	Gorgonzola	15.67	9	20

Suponga que la columna "UnitsOnOrder" es opcional y puede contener valores **NULL**.

Mire la siguiente declaración **SELECT**:

```
SELECT ProductName, UnitPrice * (UnitsInStock + UnitsOnOrder)
FROM Products;
```

En el ejemplo anterior, si alguno de los valores de "UnitsOnOrder" es **NULL**, el resultado será **NULL**.

La función **ISNULL()** de SQL Server le permite devolver un valor alternativo cuando una expresión es **NULL**:

```
SELECT ProductName, UnitPrice * (UnitsInStock +
ISNULL(UnitsOnOrder, 0))
FROM Products;
```